算法基础

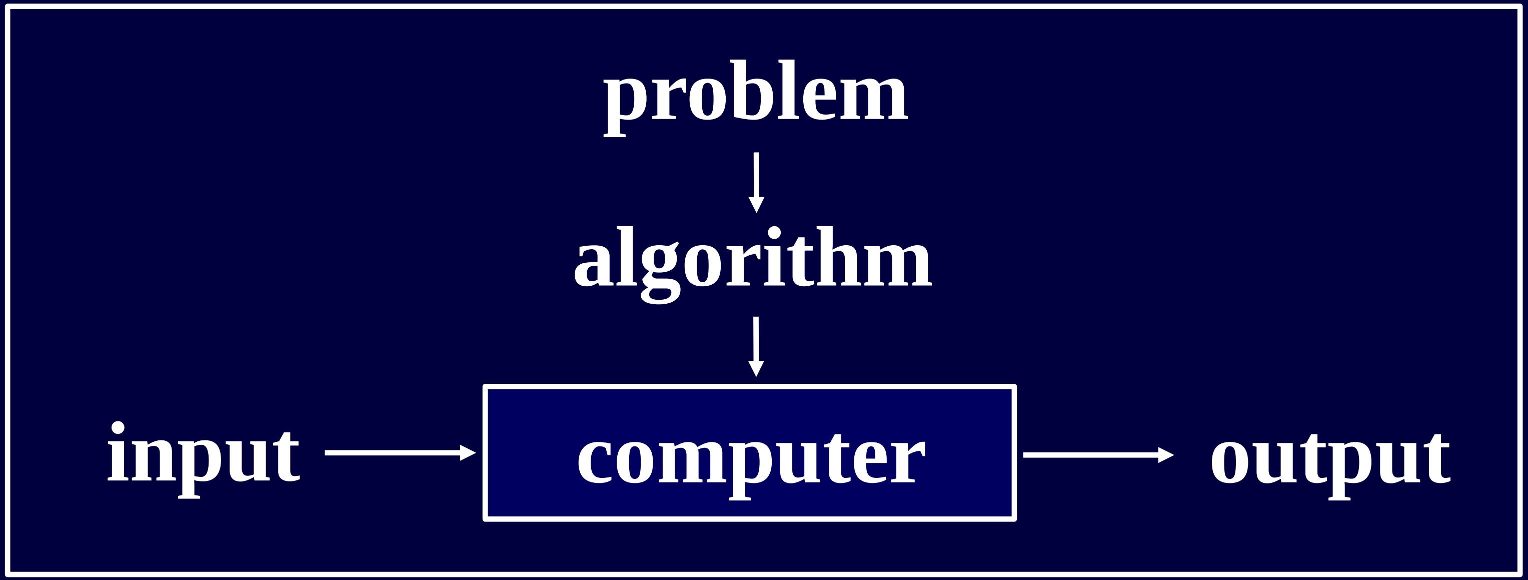
什么是算法？

- 算法(algorithm)的**非形式定义：**算法是任何**良定义(well-defined)**的**计算过程**，该过程取某个值或值的集合作为输入，并产生某个值或值的集合作为输出。换句话说，算法是**一个计算步骤的序列**，这些步骤将输入数据转换为输出结果。

或者说，算法是描述**怎样达到所期望的I/O关系**的计算过程。

什么是算法？(续)

- 算法(algorithm)的另一种定义：一个算法是用于解决一个问题的无歧义指令的序列，该执行序列在有限时间内可获得任何合法的输入所需的输出。



算法相关概念

- 问题(problem)：规定了输入和输出之间的关系，可用通用语言描述。
  - 排序问题：将一系列数按照非降顺序进行排序。

    输入(input)　：具有n个数的数列 $< a_1, a_2, \ldots a_n >$

    输出(output)：对输入数列的一个排列 $< a_1', a_2', \ldots, a_n' >$，

    使得 $a_1' \le a_2' \le \cdots \le a_n'$

- 问题实例：一个问题的实例由计算该问题解所需要的所有输入组成。
  - 排序问题的一个实例：

    input：<12, 17, 67, 41>　——　output：<12, 17, 41, 67>

算法相关概念 (续)

- **输入实例：** 问题的具体计算例子
  - 排序问题的**输入实例**：

（1）1, 2, 3, 4, 5, 6, 7.

（2）12, 42, 18, 91.

（3）93, 41, 0, 52, -1, 19, 53, 90, 34.

- **问题规模：** 算法的输入实例大小
  - 上述输入实例的**规模**分别为7、4、9。

算法相关概念 (续)

- 正确的算法：若一个算法对问题的每个输入实例，均能终止于正确的输出，则称算法是正确的。

- 不正确的算法：
  - 对某些输入实例不停机；
  - 停机时给出的不是预期的结果。

Note：不正确的算法也并非绝对无用，在不正确的算法的错误概率可控时，该算法有时是有用的 (如大素数算法)。

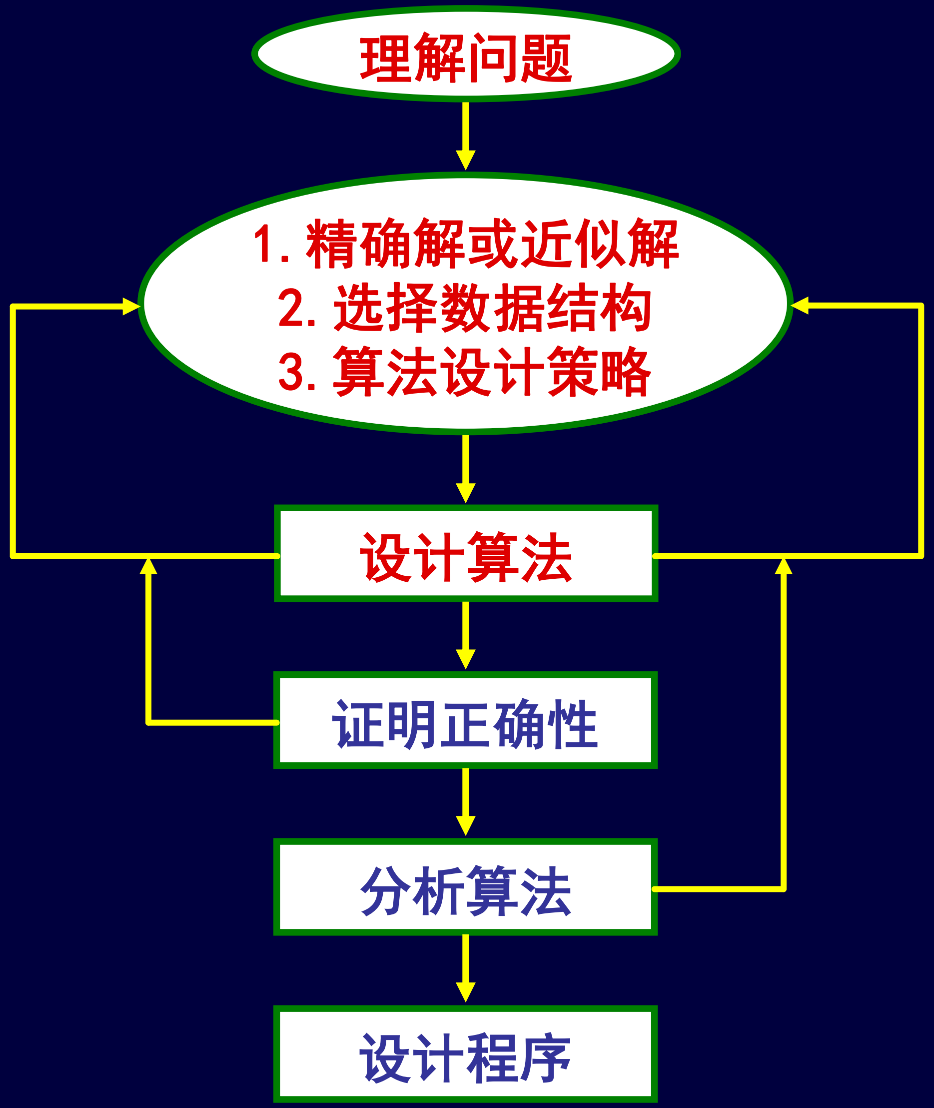
算法相关特征

- 算法(algorithm)的**特征：**
  - **输入：** 一个算法具有零或多个取自执行集合的输入值；
  - **输出：** 对每一次输入，算法具有一或多个与输入值相联系的输出值；
  - **确定性：** 算法的每一个指令步骤都是明确的；
  - **有限性：** 对每一次输入，算法都必须在有限步骤或有限时间内结束；
  - **正确性：** 对每一次输入，算法产生出正确的输出值；
  - **通用性：** 算法的执行过程可应用于所有同类问题，而不仅仅适用于特殊的输入。

算法与程序区别

- 算法(algorithm)：若干指令的有穷序列，满足输入、输出、确定性、有限性和正确性的性质。
- 程序(program)：算法用某种程序设计语言的具体实现，可以不具有有限性。
  - 操作系统是一个在无限循环中执行的程序，因而不是一个算法；
  - 操作系统的各种任务可看成是单独的问题，每一个问题由操作系统中的一个子程序通过特定的算法来实现。该子程序得到输出结果后便终止。

问题求解(Problem Solving)过程



算法的描述(Description)

- 算法的描述：可以用英语说明，可以是程序语言，只要能精确描述计算过程即可。
- 伪代码：
  - 拥有自然语言和类编程语言特性，常被用于算法描述；
  - 相较于真实代码的优势：
    1. 对特定代码的描述更加准确清晰；
    2. 不拘泥于技术细节；
    3. 体现算法本质，不受编程语言限制。

算法示例(1) —— Euclid's algorithm

问题: 寻找两个正整数m和n的**最大公约数**$gcd(m,n)$

Examples: $gcd(60,0) = 60$, $gcd(60,24) = 12$, $gcd(m,n) = ?$

欧几里得算法(Euclid's algorithm)是基于对下列等式的反复应用：

$$gcd(m,n) = gcd(n, m \bmod n),$$

当$gcd(m,n)$的第二项变为0，其第一项就成为将输出的结果。

Example: $gcd(60,24) = gcd(24,12) = gcd(12,0) = 12$

算法示例(1) —— Euclid’s algorithm (续)

- **第一种描述**

  Step 1 If $n = 0$, return m and stop; otherwise go to Step 2.

  Step 2 Divide m by n and assign the value for the remainder to r.

  Step 3 Assign the value of n to m and the value of r to n. Go to Step 1.

- **第二种描述**

$$
\begin{aligned}
&Euclid(m, n)\\
&while\ n \neq 0\ do\\
&\qquad r \leftarrow m\ mod\ n\\
&\qquad m \leftarrow n\\
&\qquad n \leftarrow r\\
&return\ m
\end{aligned}
$$

算法示例(2) —— Sieve of Eratosthenes

问题: 找出指定范围$(0, n]$内的所有素数$prime(n)$

Examples: $prime(1) = \{\}$, $prime(5) = \{2, 3, 5\}$, $prime(n) = ?$

The Description of The Sieve of Eratosthenes

$for\ p \leftarrow 2\ to\ n\ do\ A[p] \leftarrow p$
$for\ p \leftarrow 2\ to\ \lfloor\sqrt{n}\rfloor\ do$
$\quad if\ A[p] \neq 0$ // p hasn't been previously eliminated from the list
$\quad\quad j \leftarrow p^2$
$\quad\quad while\ j \leq n\ do$
$\quad\quad\quad A[j] \leftarrow 0$ // mark element as eliminated
$\quad\quad\quad j \leftarrow j + p$

Examples: $prime(20) = \{2, 3, 5, 7, 11, 13, 17, 19\}$

算法的重要性(Importance)

- 问题：对10,000,000个整数排序
  - Case 1
    算法 1：插入排序($T(n) = 2n^2$)
    计算机 A：每秒执行10^9条指令(1GHz)
  - Case 2
    算法 2：归并排序($T(n) = 50nlgn$)
    计算机 B：每秒执行10^8条指令(100MHz)
  - 耗时比较
    - Case 1: $\frac{2\times(10^7)^2 instructions}{10^9 instructions/s} = 200000s \approx 55.6h$
    - Case 2: $\frac{50\times10^7\times \, log10^7 instructions}{10^8 instructions/s} \approx 116s$